IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.	:	10/705,208	Confirmation No.	:	2941

Applicants : Aaron Joseph MCBRIDE et al.

Filed : November 10, 2003

Title : **Method and System for Programming Virtual Robots Using a Template**

Group Art Unit : 2129

Examiner : Benjamin J. BUSS

Customer No. : 28289

MAIL STOP APPEAL BRIEF – PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

TWICE-AMENDED
APPELLANT'S BRIEF UNDER 37 C.F.R. § 41.37

Sir:

        This Amended Appeal Brief is submitted to address the defective sections of the Amended Appeal Brief filed August 19, 2009, as was indicated to Appellants in the November 24, 2009 Notification of Non-Compliant Appeal Brief. This Amended Appeal Brief is timely filed without any extensions, as it is filed within one month of the mailing date of the Notification of Non-Compliant Appeal Brief.

        The headings used hereinafter and that which is set forth under each heading are in accordance with 37 C.F.R. § 41.37(c)(1).

I hereby certify that this correspondence is being electronically submitted to the United States Patent and Trademark Office on December 18, 2009.
Lisa A. Miller
(Name of Person Submitting Paper)
Signature 12/18/2009 Date

I. **REAL PARTY IN INTEREST**

The real party in interest for the application in this Appeal is the recorded Assignee, namely, Conversive, Inc. (hereinafter, "Appellant").

II. **RELATED APPEALS AND INTERFERENCES**

There are no appeals or interferences known to the Appellant or the Appellant's legal representative, which will directly affect, or be directly affected by or having a bearing on, a decision in the present Appeal.

III. **STATUS OF CLAIMS**

Claims 2-12, 16-21, 36-39 and 41 have been cancelled. Claims 1, 13-15, 22-35, and 40, having all been rejected twice, are pending in the present application. Only independent claims 1 and 35 are the subject of this Appeal. Claims 1 and 35 stand rejected under 35 U.S.C. §103 for obviousness by United States Patent Application Publication No. 2003/0163783 to Chikirivao in view of non-patent publication entitled "The Elements of AIML Style" by Wallace.

IV. **STATUS OF AMENDMENTS**

No amendments are being made or have been made to the claims after the Final Office Action dated November 14, 2008. Section VIII, below references the attached Claims Appendix which lists the currently pending claims.

V. **SUMMARY OF CLAIMED SUBJECT MATTER**

Generally, the present invention is directed to a method and system for enabling administrators with no specialized background, training, or expertise to enter information that will be incorporated into patterns and used by a Bot, thereby enabling the Bot to correctly respond to a multitude of different questions with the information the administrator provides. Specifically, the administrator can input information into a template,

wherein the information is manipulated by an editor to create a wide range of rules which are provided to an engine together with other rules of a background nature, if so desired. The rules enable the engine to provide responses to a wide range of inputs without any further programming or action on the part of the administrator. In addition, the information will be saved and can be changed or edited later by the administrator.

For purposes of this Appeal, Appellant's representative invention is set forth as a method and system in claims 1 and 35, respectively. Reference to the underlying support for each of these claims is made with respect to the specification, as filed.

Claim 1 recites a method for maintaining at least one response by an administrator in a system for autonomously processing requests, (*See* page 1, lines 4-7) comprising the steps of: providing a template to the administrator, wherein the template includes at least one field to elicit information from the administrator; (*See* page 15, lines 4-20) receiving information from the administrator into the template; (*See* page 15, lines 4-20) and making the information accessible to a rules-based program for use in providing the at least one response in reply to a request from a user, (*See* page 7, line 23 to page 8, line 10; page 16, lines 17-23) wherein the step of making the information accessible to the rules-based program saves the information as part of the template into rules, (*See* page 15, line 21 to page 16, line 9) and wherein the step of saving the information into rules includes the steps of: retrieving rules, (*See* page 16, lines 17-23) for each rule retrieved, determining whether the rule needs information, (*See* page 15, line 21 to page 16, line 9) and if the rule needs information, retrieving the information from a corresponding field in the template and inserting the information into the rule, (*See* page 15, line 21 to page 16, line 9) wherein the step of determining whether the rule needs information includes determining if either a response layer or a logic layer needs information by identifying the presence of a signifier in the response layer or the logic layer, respectively, (*See* page 17, lines 1-22) wherein the signifier is an identifier configured to call for information such that the call for information invokes a process to select the information from a corresponding field in the template so that the information will be linked to the rule, (*See* page 16, lines 10-16; page 17, lines 1-22) and wherein the logic layer is configured to choose between various responses provided by the administrator, wherein at least one of the responses is recognized by the logic layer, wherein the chosen response is the response to be used in the response layer, (*See* page 11, line 20 to page 12, line 6; page 13, lines 4-10; page 14, lines 10-22) and retrieving information

indicated as needed from a corresponding field in the template and inserting the information into the response layer or the logic layer, as called for by the signifier (*See* page 17, lines 1-22).

Claim 35 recites a computer based system that processes inputs entered by a user and provides at least one response that is maintained by an administrator (*See* page 1, lines 4-7), comprising: an interface configured to receive information from the administrator; (*See* page 8, lines 11-23) a template accessible to the administrator, wherein the template includes at least one field to elicit information from the administrator; (*See* page 15, lines 4-20) and an engine configured to: (*See* page 7, lines 3-13) make the information accessible to a rules-based program that provides the at least one response in reply to the inputs from the user; (*See* page 7, line 23 to page 8, line 10 and page 16, lines 17-23) save the information as part of the template into rules; (*See* page 15, line 21 to page 16, line 9) retrieve the rules; (*See* page 17, lines 1-22) for each rule retrieved, determine whether the rule needs information; retrieve the information from a corresponding field in the template and insert the information into the rule if the rule needs information; (*See* page 15, line 21 to page 16, line 9) determine if either a response layer or a logic layer needs information by identifying the presence of a signifier in the response layer or the logic layer, respectively, (*See* page 17, lines 1-22) wherein the signifier is an identifier configured to call for information such that the call for information invokes a process to select the information from a corresponding field in the template so that the information will be linked to the rule, (*See* page 16, lines 10-16; page 17, lines 1-22) and wherein the logic layer is configured to choose between various responses provided by the administrator, wherein at least one of the responses is recognized by the logic layer, wherein the chosen response is the response to be used in the response layer, (*See* page 11, line 20 to page 12, line 6; page 13, lines 4-10; page 14, lines 10-22) and retrieve information indicated as needed from a corresponding field in the template and insert the information into the response layer or the logic layer, as called for by the signifier (*See* page 17, lines 1-22).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

(a)   *Is claim 1 rendered obvious under 35 U.S.C. §103 by United States Patent Application Publication No. 2003/0163783 to Chikirivao in view of non-patent publication entitled "The Elements of AIML Style" by Wallace?*

(b)   *Is claim 35 rendered obvious under 35 U.S.C. §103 by United States Patent Application Publication No. 2003/0163783 to Chikirivao in view of non-patent publication entitled "The Elements of AIML Style" by Wallace?*

VII. ARGUMENT

(a) Claim 1

The Final Office Action dated November 14, 2008 (hereinafter "the final Office Action") fails to set forth a proper obviousness rejection with respect to claim 1, as the Chikirivao and Wallace references taken in combination fail to anticipate every limitation of claims 1. Reference is made herein to a Declaration Under 37 C.F.R. §1.132 submitted August 13, 2008 (hereinafter "*Declaration*") signed by Mr. Jack Keane, the Chief Technical Officer of the Assignee.

Apellants will now address various claim limitations, in turn, as they have been addressed by the Examiner beginning on page 17 of the final Office Action.

"template"

The Wallace publication simply discloses a mark-up language for inputting knowledge into chat-bots. The Wallace publication fails to disclose a "template" and/or a "logic layer" consistent with the claims as applied across both the Chikirivao publication and the Wallace publication. The Examiner acknowledges on page 5 of the final Office Action how "the term "<template>" in the Wallace publication is closer to the claimed "rules" than the claimed "template"." The claims have limitations encompassing the terms "rules" and "templates", thereby indicating that the terms are literally distinguished from each other, as they serve different roles and embody different concepts. Thus, it can be seen that an inconsistent reading of the disclosure of the Wallace publication is applied to the claims at issue.

Specifically, in §5(a) of his *Declaration*, Mr. Keane declares:

> [A]s set forth in the present application, the purpose of a template, is a way to create a specific rule or rules, based on a pre-defined form (the template), containing markers for additional information needed to define the rule (signifiers), as provided by the administrator. In the Wallace publication, AIML tags, which are equated by the Examiner with the Applicants' signifiers, serve a very different role, in that they are only activated during the execution of the AIML rules to control the flow of the program defined by those rules. AIML tags are an exclusively run-time control structure. In contrast, the signifiers in the Applicants' templates, are an exclusively compile-time structure, as they are used to construct run-time rules. There is no disclosure in the Wallace publication with respect to a mechanism that would correspond to a compilation of run-time roles from a partially-defined template. Every feature described in the Wallace publication is part of the run-time system. The use of the term "template", which appears in AIML, means something completely different, as it is the term used to describe the form of an output or reply in a rule. Therefore, it is not the case that AIML tags denote places where rules "need information", as an AIML tag denotes a control branch in the execution of the rule, which may cause recursion, output, or even external code execution.

Furthermore, on page 18 of the final Office Action, the Examiner asserts that the claimed "template" is in fact equated to the disclosed "template window" of the Wallace publication. However, even if this interpretation is proper, a consistent reading of this equated term would not make sense given that one of the claimed steps would then require "retrieving information indicated as needed from a corresponding field in the 'template interface'."

"signifier"

The Examiner also equates AIML tags with the "signifiers" set forth in the claims. However, equating the two fails to take into account the substantial differences in purpose and action between signifiers and AIML tags. Contrary to the Examiner's assertion, AIML tags do not indicate that a rule "needs information" as is claimed. Instead, AIML tags simply act as programming directives, as supported by the disclosure of the Wallace publication on page 12 reprinted below:

> More generally, AIML tags transform the reply into a mini computer program which can save data, activate other programs,
> give conditional responses, and recursively call the pattern matcher

to insert the responses from other categories.

Most AIML tags in fact belong to this template side sub language.

The Wallace publication refers to tags as a "sub language". AIML tags do not have the property that causes the information to be "linked to the rule", as is required by the claims. The signifiers of the present invention are markers that identify a specific piece of information "such that the call for information invokes a process to select the information from a corresponding field in the template so that the information will be linked to the rule" as is set forth in the claims.

On page 19, lines 9-14 of the final Office Action, the Examiner asserts how "[t]he AIML tags may indicate information is needed from another function (<srai>) or by picking from a finite set of choices (<random>). That <random> indicates choosing randomly does not indicate any lack of logic, so long as the random selection is between equally valid responses." Specifically, the Examiner attempts to rationalize how a random selection due to the "<random>" AIML tag can be interpreted as being the most logical choice, by stating that "a random selection is the most logical choice, given that there is no logical difference between the responses". Appellants fail to understand how logical decision making, as is claimed, is anticipated by this rationale.

"logic layer"

With respect to the "logic layer" limitation of the claims, Mr. Keane declares in §5(b) of his *Declaration*:

> The logic layer of the present invention implements a distinct processing step whereby a specific response can be selected from a set of equivalent responses in a principled manner, which may reference information not present in the user input to the system. Neither the Chikirivao nor Wallace publications show this feature. The Wallace publication describes AIML, however, AIML is a language without the explicit notion of the claimed logic layer, as responses in AIML proceed directly from rule input matching, or recursion on the input matching. It is incorrect to equate the "Graphmaster" of the Wallace publication to the claimed logic layer because the Graphmaster is a representation of the input space matching capabilities of a particular AIML definition set (e.g., A.L.I.C.E.). There is no explicit logic layer in AIML, as the output is determined by the user input and the rules. In the present invention, the matching of an input is only the first step to determining the appropriate response, followed by the processing of the logic layer.

The AIML model does not have an explicit logic layer, as found in the present invention. The <random> tag grouping in AIML allows an implicit variation among equivalent outputs through random choice. There is no disclosure, teaching, or suggestion in the Wallace publication of a mechanism equivalent to the claimed logic layer that is able to take additional information, if needed, and perform a defined computation that can determine the selection of an output from among a set of outputs.

In AIML, the <srai> tag allows for a recursive rule definition (See page 13 of the Wallace publication). While this allows considerable flexibility in the form of rule definition, it does not, in and of itself, introduce any functionality in the matching power of AIML that could not be replicated by finite-length, non-recursive matching rules. The <srai> tag does not provide the concept of choice over several possible outcomes *all appropriate for a particular input, but selectable by additional information **not** contained in the input.*

With respect to the "logic layer", the Examiner now broadly construes this term in the final Office Action to any branching program logic. However, the claim requires that the logic layer be configured to choose between various responses provided by the administrator. Thus, just because a response is recognized (i.e., identified) as a "response" in the Wallace publication, this action cannot be equated to the active step of making a choice between responses.

Accordingly, the claimed concept of a "template" and "logic layer" in the context of a consistent reading of the other limitations of the independent claims and the claims depending therefrom is not disclosed, taught, or suggested in any of the prior art of record. In light of the aforementioned arguments made with respect to the anticipation rejections under the Wallace publication, whose underlying anticipation teachings, now refuted, are used for rejecting at least the independent claims on an obviousness basis in view of the teaching of the Chikirivao publication, Appellants hereby respectfully request that the Examiner's obviousness rejection be reversed.

When undertaking an obviousness analysis, the Examiner is also required to take into account secondary considerations relation to applicant's invention. *Graham v. John Deere*, 383 U.S. 1, 148 USPQ 459 (1966). The Court of Appeals for the Federal Circuit stated in *Stratoflex, Inc. v. Aeroquip Corp.*, 713 F.2d 1530, 1538, 218 USPQ 871, 879 (Fed. Cir. 1983) that "evidence rising out of the so-called 'secondary considerations', such as commercial success, must always when present be considered en route to a determination of

obviousness" (*See also KSR v. Teleflex*, 550 U.S. ___ at 2). In §3 of his *Declaration*, Mr. Keane declares:

> The claimed invention was first commercialized in July of 2003. The Assignee has commercially pursued deployments that incorporate the claimed invention. With respect to enterprise deployments, the Assignee [has] offered, since as early as 2005 and through the present, implementations containing the claimed invention to various companies including PSEG (a major Northeastern U.S. electric and gas utility company), Qantas Airlines (the leading Australian air carrier), and the University of Phoenix (one of the largest higher educational institutions in the world, enrolling approximately 400,000 students). Currently, the Assignee is in contract negotiations to deploy its product through Sutherland Global Services, a company providing business process outsourcing services to Fortune 500 companies, including Dell. The estimated total sales of products and deployments that incorporate the claimed invention are at least $150,000. The foregoing information is indicative of the commercial success of the claimed invention.

As set forth in §4 of the *Declaration*, customers utilizing the claimed invention reduce their costs while improving their online customer service. As discussed in the *Declaration*, as an example, cumbersome manual processes have been replaced with automation-assisted online processes, thereby reducing errors and increasing customer satisfaction.

In the final Office Action, the Examiner asserts that the submission of the *Declaration*, has "not proven [that] the commercial success is [a] result of a direct correspondence with the claimed 'template' and/or 'logic layer' and/or 'signifier'." Contrary to the Examiner's understanding, a §1.132 Declaration is not required to address any particular claim limitations, but may be focused on the claimed invention as a whole (e.g., the benefit obtained from practicing what is claimed). In other words, the statements relating to secondary considerations in a §1.132 Declaration are not intended to go toward a showing of a correspondence between commercial success and how any one specific claim limitation relates to achieving that commercial success. Rather, the §1.132 Declaration is used to show that, assuming, arguendo, that an Examiner's obviousness combination may be defensible in light of one's arguments against obviousness, the fact that commercial success has been achieved from commercially implementing what is being claimed, is indicative of the value of the invention, thereby, overriding the §103 rejection. By dismissing a portion of the *Declaration* on improper procedural grounds, the Examiner has failed to address the merits of commercial success of the claimed invention, as discussed by Appellants, above.

(b) Claim 35

The Final Office Action dated November 14, 2008 (hereinafter "the final Office Action") fails to set forth a proper obviousness rejection with respect to claim 35, as the Chikirivao and Wallace references taken in combination fail to anticipate every limitation of claims 35. Appellants offer the same arguments herein with respect to the rejection of claim 35, as have been set forth above with respect to claim 1, above. Appellants hereby incorporate by reference the arguments made above with respect to the requirement of having the Examiner properly consider the secondary considerations set forth in the *Declaration.*

Conclusion

The present invention is a novel and non-obvious way of enabling administrators with no specialized background, training, or expertise to enter information that will be incorporated into patterns and used by a Bot, thereby enabling the Bot to correctly respond to a multitude of different questions with the information the administrator provides.

It is settled law that "[a] claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Brothers Inc. v. Union Oil Co. of California*, 2 USPQ2d, 1051, 1053 (Fed. Cir. 1987). In light of the aforementioned arguments made with respect to the anticipation rejections under the Wallace publication, whose underlying anticipation teachings, are used for rejecting at least claims 1 and 35 on an obviousness basis in view of the teaching of the Chikirivao publication, Appellants hereby respectfully request that the Examiner's improper obviousness rejection be reversed.

VIII. CLAIMS APPENDIX

A copy of the claims, as presently pending, is provided in the Claim Appendix attached hereto.

Response Under 37 C.F.R. §41.67
Amended Appellant's Brief
Application No. 10/705,208
Paper Dated: December 18, 2009
Attorney Docket No. 2966-031366

11 of 20

IX. EVIDENCE APPENDIX

Below, is a copy of a Declaration Under 37 C.F.R. § 1.132 of Mr. Jack Keane,
which was filed August 13, 2008 and acknowledged by the Examiner on page 21 of the Final
Office Action dated November 14, 2008.

Application No.: 10/705,208
Attorney Docket No.: 2966-031366

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.	:	10/705,208	Confirmation No.	:	2941
Applicants	:	Aaron Joseph MCBRIDE et al.			
Filed	:	November 10, 2003			
Title	:	Method and System for Programming Virtual Robots Using a Template			
Group Art Unit	:	2129			
Examiner	:	Benjamin J. BUSS			
Customer No.	:	28289			

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. § 1.132

1. I am a citizen of the United States of America, and a resident of Metuchen, NJ.

2. I am the Chief Technical Officer for Conversive, Inc., the Assignee of the above-identified patent application, and am therefore qualified to speak on the commercial success and industry needs that are met by the claimed invention, as well as the technical merits of the invention and the deficiencies of the prior art.

3. The claimed invention was first commercialized in July of 2003. The Assignee has commercially pursued deployments that incorporate the claimed invention. With respect to enterprise deployments, the Assignee that offered, since as early as 2005 and through the present, implementations containing the claimed invention to various companies including PSEG (a major Northeastern U.S. electric and gas utility company), Qantas Airlines (the leading Australian air carrier), and the University of Phoenix (one of the largest higher educational institutions in the world, enrolling approximately 400,000 students). Currently, the Assignee is in contract negotiations to deploy its product through Sutherland Global Services, a company providing business process outsourcing services to Fortune 500 companies, including Dell. The estimated total sales of products and deployments that incorporate the claimed invention are at least $150,000. The foregoing information is indicative of the commercial success of the claimed invention.

4. Customers appreciate the fact the claimed invention found in the products and deployments allows them to reduce cost while improving their online customer service. For example, PSEG was able to replace a cumbersome manual process whereby contractors specified changes in gas and electric service for construction projects with an automation-assisted online process, reducing errors and increasing their customer
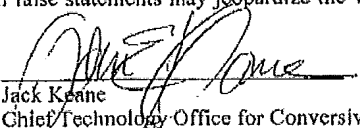
Page 1 of 3

satisfaction. PSEG is currently planning to expand their use of the Conversive products to support other online self-service processes.

     5.   The publication entitled "The Elements of AIML Style" (hereinafter "the Wallace publication") fails to disclose a "template" and/or a "logic layer", as it is described in the present application and in the context of the claims, and with respect to consistency of meaning of such terms in United States Patent Application Publication No. 2003/0163783 (hereinafter "the Chikirivao publication").

     (a)   Specifically, as set forth in the present application, the purpose of a template, is a way to create a specific rule or rules, based on a pre-defined form (the template), containing markers for additional information needed to define the rule (signifiers), as provided by the administrator. In the Wallance publication, AIML tags, which are equated by the Examiner with the Applicants' signifiers, serve a very different role, in that they are only activated during the execution of the AIML rules to control the flow of the program defined by those rules. AIML tags are an exclusively run-time control structure. In contrast, the signifiers in the Applicants' templates, are an exclusively compile-time structure, as they are used to construct run-time rules. There is no disclosure in the Wallace publication with respect to a mechanism that would correspond to a compilation of run-time rules from a partially-defined template. Every feature described in the Wallace publication is part of the run-time system. The use of the term "template", which appears in AIML, means something completely different, as it is the term used to describe the form of an output or reply in a rule. Therefore, it is not the case that AIML tags denote places where rules "need information", as an AIML tag denotes a control branch in the execution of the rule, which may cause recursion, output, or even external code execution.

     (b)   The logic layer of the present invention implements a distinct processing step whereby a specific response can be selected from a set of equivalent responses in a principled manner, which may reference information not present in the user input to the system. Neither the Chikirivao nor Wallace publications show this feature. The Wallace publication describes AIML, however, AIML is a language without the explicit notion of the claimed logic layer, as responses in AIML proceed directly from rule input matching, or recursion on the input matching. It is incorrect to equate the "Graphmaster" of the Wallace publication to the claimed logic layer because the Graphmaster is a representation of the input space matching capabilities of a particular AIML definition set (e.g., A.L.I.C.E.). There is no explicit logic layer in AIML, as the output is determined by the user input and the rules. In the present invention, the matching of an input is only the first step to determining the appropriate response, followed by the processing of the logic layer.

     6.   I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under § 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Jack Keane
Chief Technology Office for Conversive, Inc.
DATE: August 12, 2009

Page 2 of 3

X. RELATED PROCEEDINGS APPENDIX
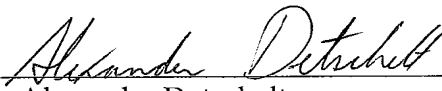
None

CONCLUSION

In view of the foregoing, it is respectfully submitted that the rejections of claims 1 and 35 under 35 U.S.C. §103(a) are improper. Accordingly, the Appellant respectfully urges the Board to reverse the Examiner's final rejections of the claims and remand the application to the Examiner with an instruction that a Notice of Allowance be issued.

A payment for $270.00 to cover the 37 C.F.R. § 41.20(b)(2) small entity fee for filing an Appeal Brief Under 37 C.F.R. § 41.37 has been previously paid in connection with filing the original Appeal Brief. The Commissioner for Patents and Trademarks is hereby authorized to charge any additional fees which may be required to Deposit Account No. 23-0650. Please refund any overpayment to Deposit Account No. 23-0650.

Any questions regarding this submission should be directed to the Appellants' undersigned representative, who can be reached by telephone at 412-471-8815.

Respectfully submitted,

THE WEBB LAW FIRM

By_____
Alexander Detschelt
Registration No. 50,261
Attorney for Appellants
700 Koppers Building
436 Seventh Avenue
Pittsburgh, PA 15219
Telephone: (412) 471-8815
Facsimile: (412) 471-4094
E-mail:webblaw@webblaw.com

CLAIMS APPENDIX

Claim 1 (Previously Presented): A method for maintaining at least one response by an administrator in a system for autonomously processing requests, comprising the steps of:

    providing a template to the administrator, wherein the template includes at least one field to elicit information from the administrator;

    receiving information from the administrator into the template; and

    making the information accessible to a rules-based program for use in providing the at least one response in reply to a request from a user, wherein the step of making the information accessible to the rules-based program saves the information as part of the template into rules, and wherein the step of saving the information into rules includes the steps of:

        retrieving rules,

        for each rule retrieved, determining whether the rule needs information, and

        if the rule needs information, retrieving the information from a corresponding field in the template and inserting the information into the rule, wherein the step of determining whether the rule needs information includes determining if either a response layer or a logic layer needs information by identifying the presence of a signifier in the response layer or the logic layer, respectively, wherein the signifier is an identifier configured to call for information such that the call for information invokes a process to select the information from a corresponding field in the template so that the information will be linked to the rule, and wherein the logic layer is configured to choose between various responses provided by the administrator, wherein at least one of the responses is recognized by the logic layer, wherein the chosen response is the response to be used in the response layer, and retrieving information indicated as needed from a corresponding field in the template and inserting the information into the response layer or the logic layer, as called for by the signifier.

Claims 2-12 (Cancelled)

Claim 13 (Previously Presented): The method according to claim 1, wherein the signifier is a tag in a text string.

Claim 14 (Previously Presented): The method according to claim 1, wherein the signifier is an instruction embedded in a text string.

Claim 15 (Previously Presented): The method according to claim 1, wherein the signifier is a code.

Claims 16-21 (Cancelled)

Claim 22 (Previously Presented): The method according to claim 1, wherein the step of retrieving rules retrieves all of the rules in a template information script.

Claim 23 (Original): The method according to claim 1, wherein the step of making the information accessible to the rules-based program is accomplished by receiving a manual command from a user.

Claim 24 (Original): The method according to claim 1, wherein the step of making the information accessible to the rules-based program is accomplished automatically upon the occurrence of a predefined event.

Claim 25 (Original): The method according to claim 24, wherein the predefined event is closing of the template.

Claim 26 (Original): The method according to claim 24, wherein the predefined event is passage of a predetermined amount of time.

Claim 27 (Original): The method according to claim 24, wherein the predefined event is activation of a save function by the administrator.

Claim 28 (Original): The method according to claim 1, further including the step of enabling the administrator to edit the information.

Claim 29 (Original): The method according to claim 28, wherein the step of enabling the administrator to edit the information includes the steps of:

retrieving the information,

posting the information in at least one appropriate field in the template,

receiving edited information from the administrator into the template, and

making the edited information accessible to the rules-based program for use in providing the at least one response in reply to a request from the user.

Claim 30 (Original): The method according to claim 29, wherein:

the step of making the information accessible to the rules-based program saves the information as part of the template, and

the step of retrieving the information includes the step of restoring the information to the at least one field.

Claim 31 (Original): The method according to claim 29, wherein:

the step of making the information accessible to the rules-based program saves the information as structured data, and

the step of retrieving the information includes the steps of, for at least one of the at least one field in the template:

retrieving instructions indicating where the information is stored, and

executing the instructions to retrieve the information.

Claim 32 (Original): The method according to claim 29, wherein:

the step of making the information accessible to the rules-based program saves the information into rules, and

the step of retrieving the information includes the steps of, for at least one of the at least one field in the template:

retrieving instructions indicating where the information is stored, and

executing the instructions to retrieve the information.

Claim 33 (Original): The method according to claim 29, wherein:

the step of making the information accessible to the rules-based program saves the information into rules, and

> the step of retrieving the information includes the steps of, for each rule used:

> > determining whether the rule includes a signifier, and

> > if a signifier is included, executing instructions from the signifier to retrieve

the information associated with the rule.

Claim 34 (Original): The method according to claim 29, wherein:

the step of making the information accessible to the rules-based program saves the information into rules, and

> the step of retrieving the information includes the steps of, for each rule used:

> > determining whether the rule includes a signifier, and

> > if a signifier is included, retrieving the information tagged in the rule.

Claim 35 (Previously Presented): A computer based system that processes inputs entered by a user and provides at least one response that is maintained by an administrator, comprising:

> an interface configured to receive information from the administrator;

> a template accessible to the administrator, wherein the template includes at least one field to elicit information from the administrator; and

> an engine configured to:

> > make the information accessible to a rules-based program that provides the at least one response in reply to the inputs from the user;

> > save the information as part of the template into rules;

> > retrieve the rules;

> > for each rule retrieved, determine whether the rule needs information;

> > retrieve the information from a corresponding field in the template and insert the information into the rule if the rule needs information;

> > determine if either a response layer or a logic layer needs information by identifying the presence of a signifier in the response layer or the logic layer, respectively, wherein the signifier is an identifier configured to call for information such that the call for information invokes a process to select the information from a corresponding field in the

template so that the information will be linked to the rule, and wherein the logic layer is configured to choose between various responses provided by the administrator, wherein at least one of the responses is recognized by the logic layer, wherein the chosen response is the response to be used in the response layer, and

retrieve information indicated as needed from a corresponding field in the template and insert the information into the response layer or the logic layer, as called for by the signifier.

Claims 36-39 (Cancelled)

Claim 40 (Original): The computer based system according to claim 35, further including an editor adapted to access the information and enable the administrator to edit the information.

Claim 41 (Cancelled)